

## MODUL PRAKTIKUM III

### PENGENALAN *CLASS*, *OBJECT*, DAN *METHOD*

#### A. Tujuan

1. Mahasiswa mampu membuat *class* dan *object*
2. Mahasiswa mampu mendeklarasikan atribut, *constructor* dan *method* pada *class*

#### B. Latihan

##### Latihan 1: Data Pribadi Mahasiswa

- a. Implementasikan UML class diagram berikut dalam Java

Mahasiswa
+ nim : int
+ status_aktif : boolean
+ <u>Siswa (i : int, status : boolean)</u>

```
public class Mahasiswa {
    public int nim;
    public boolean status_aktif;
    public void Mahasiswa(int i, boolean status){
        nim = i;
        status_aktif = status;
    }
}
```

- b. Buat *class* TesMahasiswa.java untuk membuat object dari kelas di atas.

```
public class TesMahasiswa {
    public static void main(String[] args) {
        Mahasiswa mhs1 = new Mahasiswa(1234, true);
        Mahasiswa mhs2 = new Mahasiswa (1212, false);
        Mahasiswa mhs3 = new Mahasiswa (1113, true);

        //Mencetak data mahasiswa 1
        System.out.println("Mahasiswa dengan NIM "+mhs1.nim+
            " memiliki status aktif = "+mhs1.status_aktif);
        //Mencetak data mahasiswa 2
        System.out.println("Mahasiswa dengan NIM "+mhs2.nim+
            " memiliki status aktif = "+mhs2.status_aktif);
        //Mencetak data mahasiswa 3
        System.out.println("Mahasiswa dengan NIM "+mhs3.nim+
            " memiliki status aktif = "+mhs3.status_aktif);
    }
}
```

c. Output program di atas adalah:

d. Bagaimana perubahan *sourcecode* dan *class diagram* dari *class* Mahasiswa.java apabila ditambahkan atribut nama dan jurusan pada *class* Mahasiswa.java.

e. Bagaimana perubahan *sourcecode* dari *class* TesMahasiswa.java untuk menampilkan semua atribut dari object mhs1, mhs2, dan mhs3.

f. Output program di atas adalah:

### Latihan 2: Data Persewaan Mobil

a. Implementasikan UML class diagram berikut dalam Java

Mobil
- id_mobil : int - merk : String - kapasitas : int - harga_sewa : int - <u>Mobil (no : int, tipe : String, jumlah : int, harga : int)</u>
+ setIdMobil (i : int) + getIdMobil () : int + seMerkMobil (i : String) + getMerkMobil () : String + setKapasitasMobil (i : int) + getKapasitasMobil () : int + setHargaMobil (i : int) + getHargaMobil () : int

```

public class Mobil {
    private int id_mobil;
    private String merk;
    private int kapasitas;
    private int harga_sewa;
    public Mobil(int no, String tipe, int jumlah, int harga) {
        id_mobil = no;        merk = tipe;
        kapasitas = jumlah;    harga_sewa = harga;
    }
    public void setIdMobil(int i) {
        id_mobil = i;
    }
    public int getIdMobil() {
        return id_mobil;
    }
    public void setMerkMobil(String i) {
        merk = i;
    }
    public String getMerkMobil() {
        return merk;
    }
    public void setKapasitasMobil(int i) {
        kapasitas = i;
    }
    public int getKapasitasMobil() {
        return kapasitas;
    }
    public void setHargaMobil(int i) {
        harga_sewa = i;
    }
    public int getHargaMobil() {
        return harga_sewa;
    }
}

```

- b. Buat *class* TesMobil.java untuk membuat object dari kelas di atas.

```

public class TesMobil {
    public static void main(String[] args) {
        Mobil m1 = new Mobil(1, "Daihatsu", 2, 150);
        Mobil m2 = new Mobil(2, "Nissan", 4, 250);
        Mobil m3 = new Mobil(3, "Honda", 7, 200);

        //Mencetak data mobil 1
        System.out.println("Mobil "+m1.getIdMobil()+
            " dengan merk "+m1.getMerkMobil()+" memiliki kapasitas "+
            m1.getKapasitasMobil()+" dengan sewa per jam-nya sebesar "+
            m1.getHargaMobil()+"K");
        //Mencetak data mobil 2
        System.out.println("Mobil "+m2.getIdMobil()+
            " dengan merk "+m2.getMerkMobil()+" memiliki kapasitas "+
            m2.getKapasitasMobil()+" dengan sewa per jam-nya sebesar "+
            m2.getHargaMobil()+"K");
        //Mencetak data mobil 3
        System.out.println("Mobil "+m3.getIdMobil()+
            " dengan merk "+m3.getMerkMobil()+" memiliki kapasitas "+
            m3.getKapasitasMobil()+" dengan sewa per jam-nya sebesar "+
            m3.getHargaMobil()+"K");
    }
}

```

c. Output program di atas adalah:

d. Apa yang terjadi apabila pada *class* TesMobil.java user memanggil atribut *object* secara langsung tanpa melalui *method* set dan get sebagaimana *sourcecode* di atas.

e. Bagaimana perubahan *sourcecode* dan UML *class diagram* dari *class* Mobil.java apabila ditambahkan *method* untuk mencetak semua atribut dari *object* yang bertipe Mobil.

f. Modifikasi *sourcecode* TesMobil.java untuk menampilkan ulang atribut *object* yang bertipe Mobil dengan *method* yang telah dibuat sebelumnya.

### C. TUGAS PRAKTIKUM

1. Transformasikan UML *class diagram* berikut ke dalam Java program. Jalankan file LatihanPraktikum1.class. Jika tampilan di layar tampak seperti dibawah ini, maka program anda sudah benar. Jika tidak sama, benahi kembali program anda dan lakukan hal yang sama seperti diatas.

Tabungan
- saldo : int
+ <u>Tabungan(initsaldo : int)</u>
+ getSaldo() : int
+ simpanUang(jumlah : int)
+ ambilUang(jumlah : int) : boolean

Output:

```
Saldo awal : 5000
Jumlah uang yang disimpan : 3000
Jumlah uang yang diambil : 6000 ok
Jumlah uang yang disimpan : 3500
Jumlah uang yang diambil : 4000 ok
Jumlah uang yang diambil : 1600 gagal
Jumlah uang yang disimpan : 2000
Saldo sekarang = 3500
```

2. Transformasikan UML *class diagram* berikut ke dalam Java program. Jalankan file LatihanPraktikum2.class untuk menambahkan muatan sebanyak keinginan user.

Truk
- muatan : double
- muatanmaks : double
+ <u>Truk(beratmaks : double)</u>
+ getMuatan() : double
+ getMuatanMaks() : double
+ tambahMuatan(berat : double) : boolean

Output:

```
Muatan maksimal = 1000
Muatan saat ini = 0
Apakah user ingin menambahkan muatan? Y

Tambahan muatan = 500
Muatan berhasil ditambahkan.
Muatan saat ini 500.
Apakah user ingin menambahkan muatan? Y

Tambahan muatan = 700
Tambahan muatan melebihi kapasitas maksimal. Muatan tidak berhasil ditambahkan.
```

Muatan saat ini 500.  
Apakah user ingin menambahkan muatan? T

#### D. TUGAS RUMAH

1. Seorang promotor ingin membuat konser peringatan hari kemerdekaan RI. Oleh karena itu promotor tersebut menugaskan pegawainya untuk mengidentifikasi pengisi acara yang akan diundang. Hasil identifikasi dicatat dalam suatu tabel berikut:

Obyek	A	B	C	D	E
Artis1	MC	Pria	Pesawat	1	1500
Artis2	MC	Wanita	Travel	1	700
Artis3	Penyanyi	Wanita	Pesawat	1	5000
Artis4	Band	Campuran	Kereta api	5	10000

Bantulah promotor tersebut dalam membuat *sourcecode* dan UML *class diagram* Artis.java. Tambahkan method infoArtis() yang bertujuan untuk menampilkan semua karakteristik Artis. Buatlah kelas TesArtis.java yang berisi pembuatan dan pengesetan karakteristik 4 (empat) buah obyek bernama Artis1, Artis2, Artis3, Artis4 serta menampilkan info karakteristik artis.